Submitted to IZS 90 *"Electronic circuits and systems for communication"* March 5-8,1990, ETH Zurich

Design of an enhanced FDDI-II System

M. Siegel, K. Sauer, W. Schödl, M. Tangemann
Institute of Communications Switching and Data Technics
University of Stuttgart
Seidenstr. 36
D-7000 Stuttgart 1
Telephone: +49 711 121-2474
Telefax: +49 711 121-2477
E-Mail:SIEGEL@nvdv.e-technik.uni-stuttgart.dbp.de

Abstract

The growing demand for more bandwidth and integrated telecommunication services in the customer premises area can be satisfied with the hybrid high speed local area network FDDI-II (Fibre Distributed Data Interface). This paper presents a hardware design based on existing highly integrated components for the FDDI protocol. The design contains all features according to the proposed FDDI-II standard. Additionally our design provides features for enhanced performance, higher reliability and extensive support for network management. The central component which is called hybrid multiplexer (H-MUX), is described in detail. Implementation aspects of the station design, strategies of interconnecting the FDDI-II system to public networks, especially to Integrated Broadband Communication Networks (IBCN) are provided. Furthermore the feasibility of current CMOS technology and design strategies for the high performance requirements of this communication system are discussed.

Introduction

In the field of data communications new concepts are evolving. The so-called High Speed Local Area Networks (HSLANs) provide features such as high transmission bandwidth (greater than 100 Mbps), they are extended to large geographical areas (medium length up to 100 km) and they support different traffic types (circuit and packet switched traffic). With

these properties they are well-suited for future inhouse communication requirements. HSLANs can be used as front end, backbone or back end networks.

Up to now, various types of HSLANs have been developed, which can be distinguished by the topologies they are based on and the applied Media Access Control (MAC) protocol. The Distributed Queue Dual Bus (DQDB) proposal /6/ e.g. employs two unidirectional buses flowing in opposite directions and the MAC-protocol for packet switched traffic uses a completely distributed request mechanism. Other representatives are the Cambridge and Orwell Rings /13, 14/. Both are based on slotted ring topologies. Two HSLAN proposals have entered the international standardization process, DQDB and the Fibre Distributed Data Interface (FDDI).

In this paper we are focusing on the FDDI system /1-5,8/ which provides a transmission bandwidth of 100 Mbps on fibre optic loops. For reasons of fault tolerance a dual ring topology has been chosen. The maximum ring length is 100 km and up to 500 nodes can be connected to the ring. The standardization of FDDI proceeds in two steps. FDDI-I /1-3/ was approved as an American National Standard in 1988. Providing only packet switched services two different access schemes can be distinguished, namely the synchronous and asynchronous traffic class. The proposal of the FDDI-II standard /5/ is a superset of FDDI-I which additionally provides an isochronous traffic class.

In this paper a node design of an enhanced FDDI-II system is presented. The main characteristics of the proposed FDDI-II system are described in the next section. The detailed node design is described in section 3. Section 4 highlights the major enhancements compared with the standardized FDDI-II version. The description of the most important building block, the so-called Hybrid Multiplexer, is presented in section 5. Finally some implementation aspects will be discussed and an outlook to further extensions will be given.

FDDI-II Basics

FDDI is a protocol employed for a 100 Mbps token passing physical ring using a fibre optic medium. In the year 1986 draft proposals of the Physical /1, 2/ and in 1988 an ANSI standard of the MAC layer /3/ of the FDDI protocol were published which describe only a non-isochronous operation mode.

At the moment a hybrid mode, so-called FDDI-II, is defined. This hybrid mode is a superset of the FDDI standard and supports isochronous traffic as well as non-isochronous traffic /5, 8/. This paper is focused on an implementation based on the FDDI-II protocol with enhanced performance features and the same guaranteed reliability as described by the ANSI standard.

The FDDI-II protocol supports three traffic classes, where only the so-called isochronous traffic class can carry the circuit switched (CS) services. The other two classes, which are defined in the FDDI MAC standard /3/, handle the packet switched (PS) traffic. The non-isochronous traffic classes are subdivided in one synchronous traffic class which allows data transmission by a pre-allocated bandwidth scheme and an asynchronous traffic class. The data transmission in the asynchronous traffic class is controlled by a timed token protocol using a so-called non-restricted token. In this non-restricted token mode up to eight priority levels can be recognized. A special restricted token mode in the asynchronous traffic class allows dialogue oriented connections between some selected stations.

To provide both circuit switched and packet switched capabilities, a hybrid switching scheme for FDDI-II is used. This switching scheme is based on a cycle structure which is generated from a cycle master station periodically every 125 ms (Fig. 1).

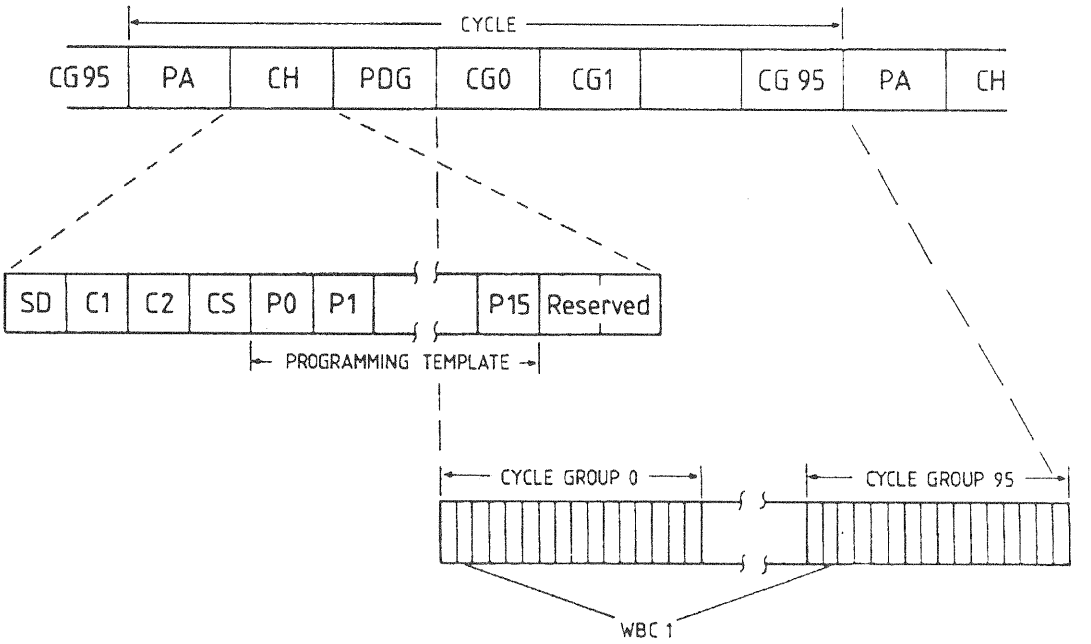

Figure 1: FDDI-II Cycle Structure

Every cycle contains a cycle header, a dedicated packet data group and 96 cycle groups. The packet data group provides a guaranteed minimum bandwidth of 768 kbps for the packet switched services. The bandwidth used by the cycle groups is partitioned into 16 so-called Wide Band Channels (WBC's), each with a bandwidth of 6.144 Mbps. Each of these WBC's is individually assigned to either packet or circuit switched traffic according to the programming template of each cycle header. The WBC's and the dedicated packet data group are arranged in a byte-interleaved manner across the 96 cycle groups.

The network management can allocate channels with different bandwidth within the isochronous WBC's on the basis of 64 kbps circuit switched channels. The packet switched channel is built up by concatenating all WBC's which are associated to the PS traffic and the packet data group. The channel access to this allocated packet bandwidth is controlled by a timed token protocol.

To gain access to the media according to the timed token protocol every station contains some internal timers. One is called Token Rotation Timer (TRT). This timer measures the token inter-arrival time between the last token arrival and the actual token arrival. During the ring initialization phase a Target Token Rotation Time (TTRT) is negotiated. If the measured TRT is less then the negotiated TTRT the station gets a usable token and is able to transmit several packets; otherwise, the station must release the received token at once without transmitting any packet.

Besides the Token Rotation Timer all stations have a Token Hold Timer (THT), which measures the remaining time between the TRT and the TTRT. Exceeding this maximum transmission time the station must release the token after completing the current frame transmission.

Within the asynchronous non-restricted token class 8 priority levels are supported by using different timer thresholds. If the transmission of all pending packets of a higher priority level is finished within the token hold time, lower priorities are able to transmit data packets if their threshold is greater than the remaining token hold time.

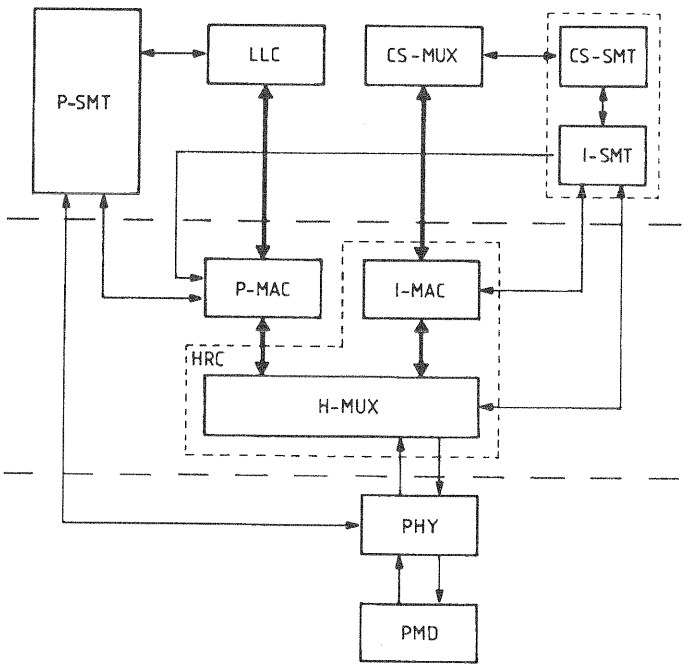

Figure 2: Basic Functional Model of a FDDI-II Station

The basic functional model given in Fig. 2 shows the different parts of a FDDI-II station. The parts PMD, PHY, P-MAC, P-SMT are recommended in the documents /1 - 4/, whereas the Hybrid Ring Control is described in /5/. The incoming data stream according to the frame structure given in Fig. 1 is switched in each station to the P-MAC or I-MAC by the Hybrid Multiplexer (H-MUX). This switching function is controlled by analysing the information of the programming template of each cycle in real-time.

Within the FDDI standard a dual ring configuration is mandatory to guarantee high reliability. Under normal operation only one ring is used; the second ring serves as a spare ring to allow reconfiguration after a physical ring failure. One of our ideas is to use both rings for data transmission under normal operation to increase the system bandwidth. In the following sections the design aspects and some enhancements of the basic FDDI-II functions in terms of reliability and performance are described and details of our implementation are discussed.

Node Design

Our design is build around an available FDDI-I chipset of Advanced Micro Devices (AMD), which AMD calls the SUPERNET Family /7/. This concept demonstrates that FDDI-II is not only a logical superset of FDDI but also that a hardware upgrade from existent FDDI-implementations towards FDDI-II is feasible. Extending existent FDDI-I installations just with an add-on hardware is driven by the actual market situation which is in some kind self-blocking. Investments into FDDI-II seems to be less risky after FDDI-I has shown sufficient operability and reached wide acceptance. The drawback is that many customers prefer waiting for FDDI-II instead of installing FDDI-I now.

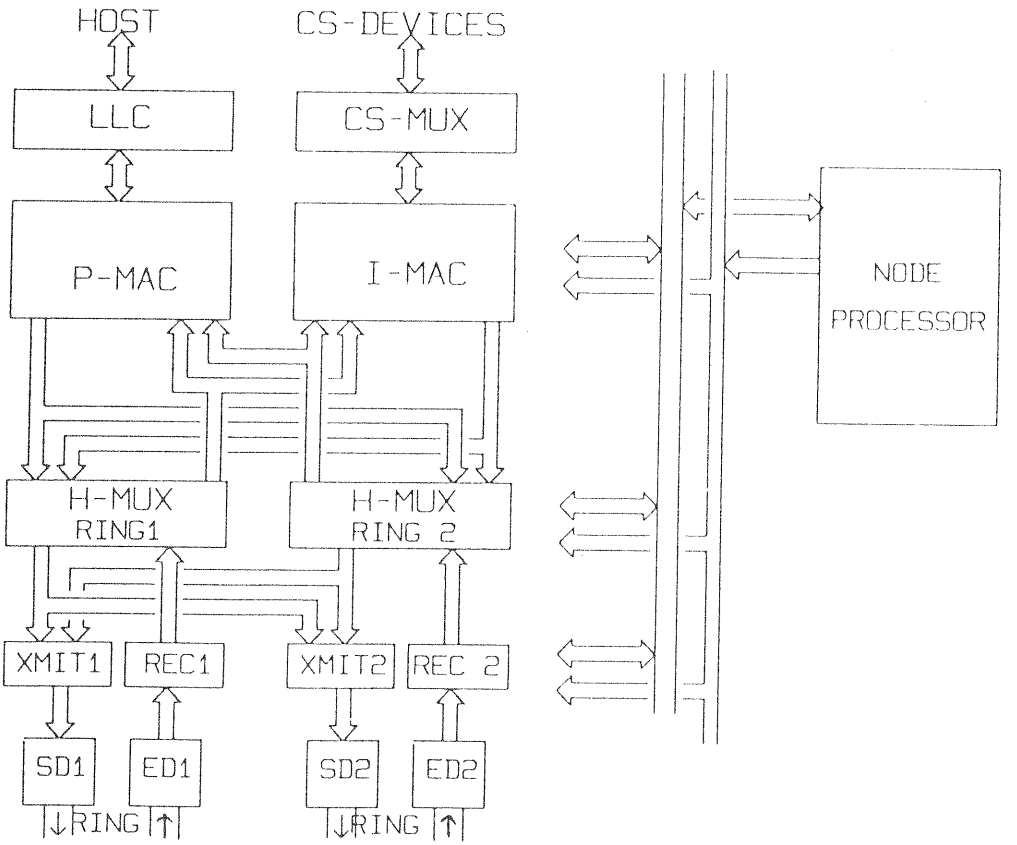

Figure 3: Dual Attached FDDI-II Station

Fig. 3 shows our design for a FDDI-II dual attached station (DAS), operating in a way matching the standard. This means one ring is active and the second is in the hot standby mode.

The blocks of the diagram and their functions are described below. The optic/electric conversion and vice versa is done in the SD and ED blocks, respectively. The physical layer functions are realized within the two blocks ENDEC (Encoder/Decoder) and EDS (Endec Data Separator) which are part of the AMD chipset /9/.

Covered functions are:

- Clock recovery and generating
- Loopback circuit to support device testing
- Appropriate conversion of line and symbol codes
- Serial to parallel conversion according to the border of the bytes defined by the starting delimiter
- Smoothing function, i.e. shorten or lengthen the preamble
- Decoding and indication of the line state in receive direction, generating the line state in transmit direction.

Besides the mentioned basic functions there are some hardware specific features like loopback, repeat filter and electrical by-pass mechanisms to assist device testing as well as support for station management and fault tolerance.

Inside the Configuration Switch the routing of one of the two data input streams takes place. The control of the switch is performed by the station management to allow different ring configurations.

On the first view the basic functionality of the H-MUX seems to be only the switching of the incoming byte stream according to the programming template to the MAC-layer for either PS or CS in receive direction and vice versa in transmit direction. But to achieve this functionality a lot of additional tasks are required. These tasks must be supported by the station management. The H-MUX is described in detail in a following chapter.

Above the H-MUX the packet MAC represented by the FORMAC (Fibre Optic Ring Media Access Control) chip of the Supernet family and the isochronous MAC block are located. The packet MAC is responsible for handling all asynchronous traffic according to the timed token protocol. Therefore it contains internal timers for the TRT and THT, an overall valid trans-

mission timer TVX and registers for specific protocol parameters as well as the address detection.

From the diagnostic point of view the FORMAC offers several internal loopback modes and control signals for maintenance. A disadvantage of the chip is that many functions, especially for station management, must be supported outside the chip with extra hardware.

The isochronous MAC (I-MAC) controls the access to the bytes of WBCs which belong to the CS-connections of that station. This means that the I-MAC has to manage a steering map containing information for all 1536 data bytes of one cycle (96 cyclic groups with one byte of every wideband channel each).

The CS-Multiplexer (CS-MUX) maps the isochronous data stream according to the steering map of the I-MAC to the connections allocated by the CS users. Thus CS channels on the basis of n*64 kbps can be switched by concatenation. An allocation of less than 64 kbps is also possible.

Memory control and data transfer which found the base for most upper protocol functions are implemented by a data path controller (DPC) and RAM buffer controller (RBC) /9/.

Together with a node processor the described blocks represent all necessary components for a FDDI-II node. A direct communication between node processor and each block is required. In Fig. 3 these communication paths are depicted by the horizontal arrows; the data flow is given by vertical arrows. Some investigations are currently under work to evaluate multi-processor architectures for the station and system management.

Enhancements

The FDDI-II system design presented herein contains some features, which increases the capabilities of the FDDI-II draft international standard without loosing compatibility. The main features are listed below :

- Full usage of both rings under normal working conditions
- Two additional by-pass mechanisms
- Further security mechanisms are implemented
- Hardware support for network management functions
- Ring modes (FDDI-I or FDDI-II) can be handled individually for both rings
- Interworking of FDDI-II and ATM based public networks is possible

Within the standard one of the two rings is proposed only as a spare ring. This is a wastage of bandwidth and of hardware resources. The design we have chosen avoids this by using both rings under normal operating conditions. This includes the use of one ring in the FDDI-I mode and the other in the hybrid mode simultaneously.

The capability to reconfigure to a single folded ring in case of failures is not lost. In principle the hardware offers two ways of reconfiguration. The first is to switch to the lowest mode, which is common to all connected stations on both rings, e.g. the basic mode (FDDI-I). This lowest common level must be fixed during the procedure of ring initialisation. The actual task in case of a physical ring failure is to rearrange only the two stations closest to the damage. The second possibility requires internal reconfiguration of all stations, which do not support two FDDI-II rings. These stations have to be by-passed and can not participate in data transmission until the ring is healed or the reconfigured ring changes to the basic mode. The result is again one functional ring, which is able to carry hybrid traffic. Obviously, the management of the latter way is more difficult.

The standard proposes optical by-passes (optical relais) on the physical layer which enable ring operation even with some non-active stations connected to the ring. Disconnecting an attached station from ring causes a reconfiguration of the whole system and loss of data independent of the ring mode. The physical readability of the optical signals could also decrease due to damping effects. Therefore we developed additional electrical by-passes, which insert exactly the same delay to the ring as they bridge. This means that it is possible to remove and insert stations without changing the ring length and thus causing data and synchronization losses and forcing restart operations. In order to avoid problems of token losses the activating and deactivating of the by-passes can only be done if the station has not captured the token.

From the security point of view control mechanisms are implemented in hardware to avoid disallowed access. Two types of consistency checks are performed by comparing the user access requests with the actual channel allocation on the ring. These consistency checks support data integrity as well as data security.

The constraints of real-time network management functions are supported directly by counters, comparators and with additional internal communication paths.

The main enhancement is the possibility to use both rings, if all connections are operating normally, and thus actually double the available bandwidth without loosing the fault tolerant characteristic of a double ring system.

The station hardware to support such a concept is presented in Fig. 4. It is obvious that the management in that configuration has at least double the volume of the simple standard configuration of Fig. 3, which is similar to the standard proposal.

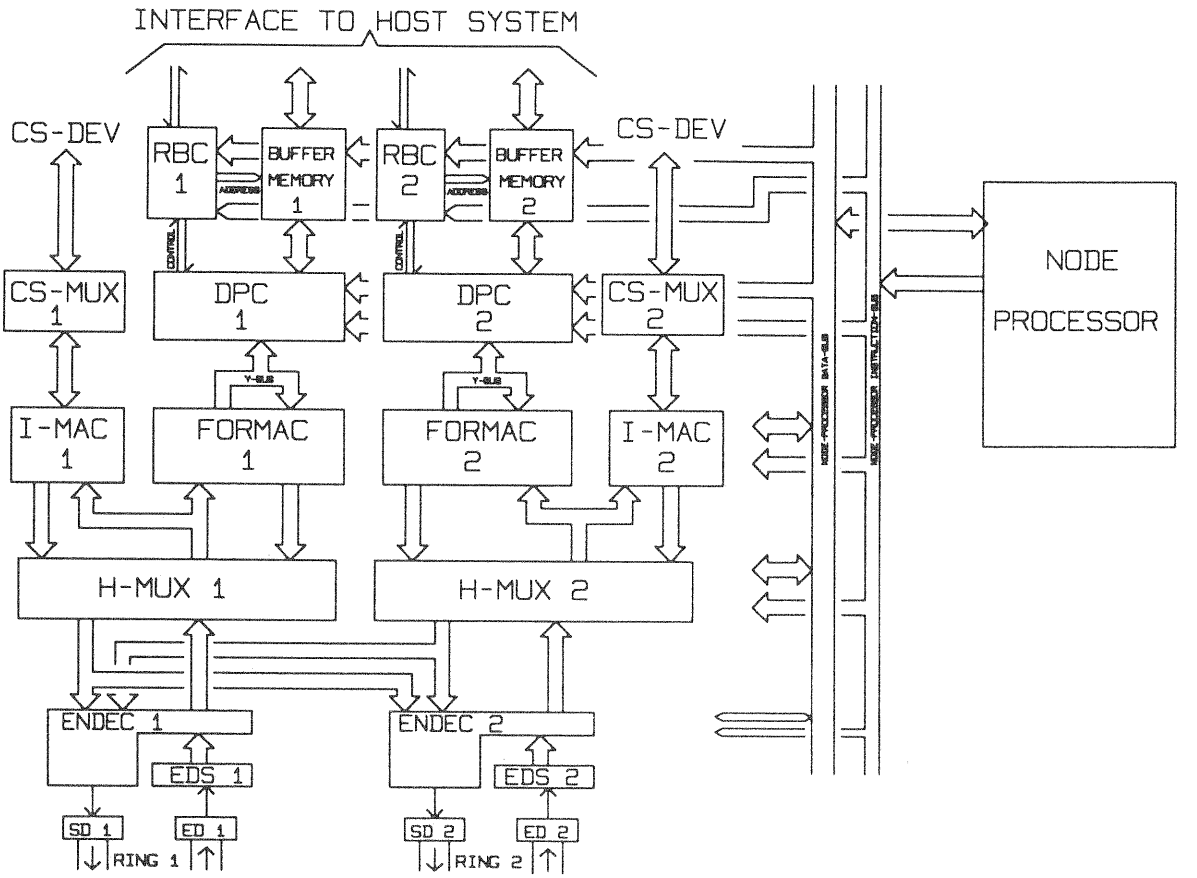

Figure 4: Enhanced DAS Design

Hybrid Multiplexer

Most of the enhancements mentioned above directly influence the design of the H-MUX. Thus the hybrid multiplexer becomes the central element in our FDDI-II design. In this chapter first the basic functions of the H-MUX are described. Then the special requirements for the H-MUX of a monitor station are shown (cf. Fig. 5).

The FDDI-II protocol needs one master station. Due to fault tolerance reasons, more than one station - ideally all - should have the capability of performing master functions. These stations are called monitor stations. To avoid the development of two different designs a maximum subset of identical functions of monitor/master and pure so-called slave stations has been worked out and a feasible interface of that subset was defined in such a way, that it is now

possible to just add the monitor specific hardware module to every slave station. Fig. 5 shows the general structure and the significant interface signals of the H-MUX.

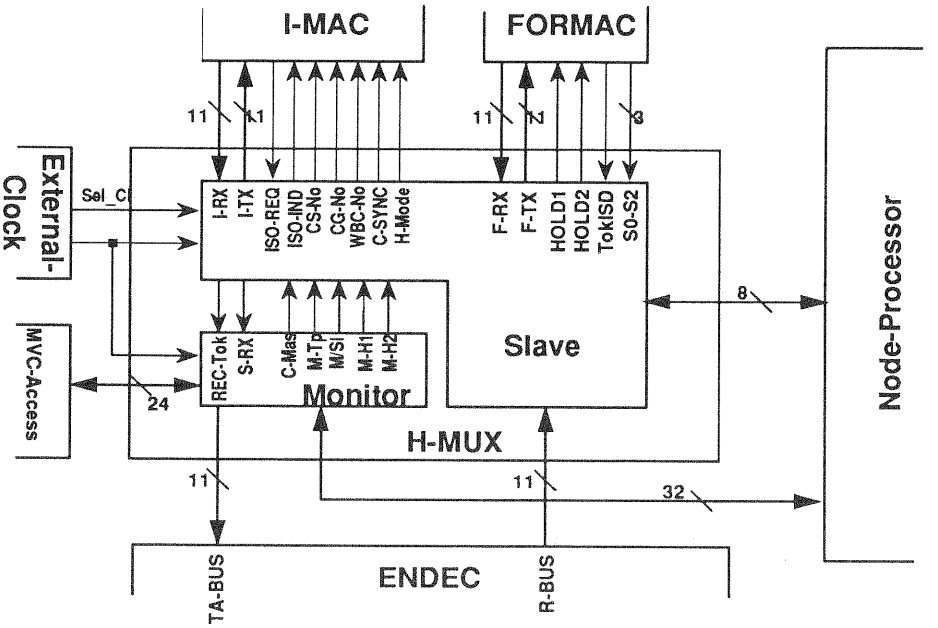

Figure 5: Block Diagram of the H-MUX

The ability of suppressing the cycle control symbols at the beginning of a frame directly in hardware offers a way to disable hybrid mode. Due to the implementation in hardware this is a very fast way to finish the request procedure for mode changing and also enables every station to switch from hybrid to basic ring mode without the need of complex and time-consuming software mechanisms.

Beside the obvious demultiplexing of the incoming data stream to the P-MAC or I-MAC unit according to the programming template and vice versa some realized slave H-MUX functions out of the standard proposal are :

- Detection of the actual operation mode of the ring
- Detection of the starting-delimiter - a sequence of a J and a K symbol - in both possible ring modes. (in basic mode this is the start of a packet, in hybrid mode it is the start of a cycle)
- Decoding and real-time handling of the programming template in hybrid mode
- Error handling in the programming template
- Communication interface to station management
- Sequence and synchronization error detection
- Interworking with the node processor

Our presented design includes further functionalities:

- Detection and handling of all FDDI-II error and reconfiguration procedures
- Insertion and removal of station by-passes without changing the ring length
- Detection of token capture within H-MUX
- Capabilities to increase data security
- Features for improved data integrity
- Enhanced hardware support for station management

A coarse logical division of the slave-H-MUX functionality results in Fig. 6. Around a central H-MUX control unit (HCU) there are specialized control units for receiving (RCU) and transmitting (TCU) as well as for the two different traffic types based on either isochronous circuit switching (ICU) or asynchronous packet switching (PCU).

A fault tolerant handling of errors in the programming template is part of the design of the H-MUX by means of saving the last received programming template and after detection of errors (i.e. no valid S or R symbol) taking the old values instead of the actual ones.

A monitor station must not only take care for the station itself but must also be responsible for functions which are prior conditions for the overall ring operability. The most important task

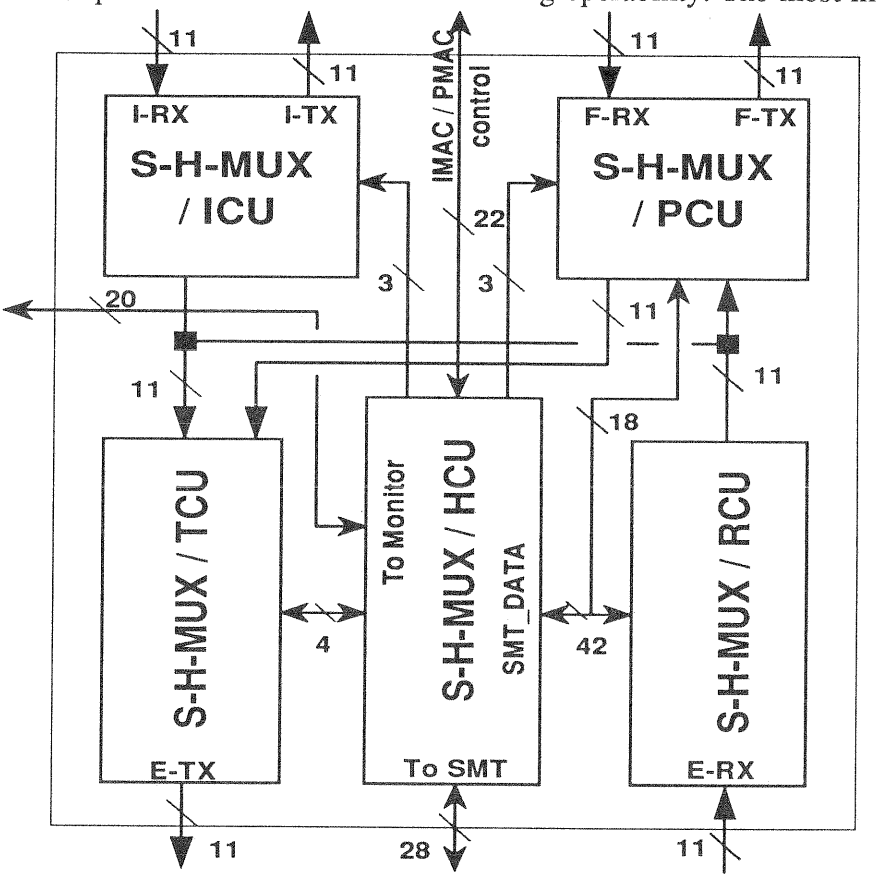

Figure 6: Main blocks of a slave H-MUX

is the generation and management of the complete cycle structure on the ring including a highly precise cycle clock generation and the complete control over the latency adjustment buffer to guarantee a physical ring length, which must always be an integer multiple of a complete cycle.

Changes in the programming template are only allowed inside a master station. Mechanisms to handle these changes and to supervise the consistency of the template in transmit and receive direction therefore are part of the monitor hardware.

Recognizing that a monitor station is the actual master as well as active participation in the diverse initialization processes are self-evident monitor tasks. Special features of our implementation are performing access to the 64 kbps management voice channel (MVC) and monitoring of the actual master station by the other monitor stations which request a new bidding process in case of inconsistencies.

Implementation Aspects

The use of chips which are not especially built for FDDI-II introduced some problems. Most of them could be solved in a standard conforming way with additional hardware and partially with deviations. A few minor details could only be implemented with not fully standard compatible solutions. Due to the fact that even the AMD chipset itself does not fully support the FDDI-I standard, it is obvious that the same problems occur at least in the PS-part of the FDDI-II design.

The main difference between the AMD supernet family and the FDDI-I standard is the support of only one priority class - instead of 8 in the standard - in the non-restricted asynchronous mode. Another major difference is that 48 bit MAC group addresses are not supported.

In our opinion the AMD chipset has some disadvantages: the late-counter and most of the timers are not accessible externally, which would be helpful for the detection of bottle-necks and for obtaining an information about the load condition of the system. Some of the counters and registers which support the management are not implemented. However there are at least control lines to trigger external hardware. The buffer concept of the AMD chipset is based on a single queue. The buffer management requires a strictly sequential arrangement of data.

The ENDEC has the feature of pre-filtering invalid from valid incoming symbols. For the basic FDDI-mode this is a welcome additional stage of early error detection, but unfortunately FDDI-II uses some of these "invalid" symbols which therefore can not pass the ENDEC. We

decided not to develop our own ENDEC, because this would need circuits with speed requirements far above the ability of CMOS-VLSI technology, for which we have experience and appropriate design tools /11/.

The basic mode only requires one kind of starting delimiter, namely one for every new frame (packet), the hybrid mode on the other side must be able to offer a second starting flag for signalling the 125 µs cycle start. The standards propose a symbol sequence J, K for the FDDI-I frame delimiter as well as the FDDI-II cycle start flag. An I, L sequence shall be used as *in cycle starting delimiter* in hybrid mode, i.e. for signalling the start of a frame inside the packet channel. The FORMAC chip intends and accepts only a J, K sequence as starting delimiter of a new frame. A conversion of the *in cycle delimiter* therefore is necessary in our design anyway even when the proposed I, L sequence would be used. To this conversion unit a programmable register was added to allow the free choice of the symbol sequence, which actually is defined as *in cycle starting delimiter*. With the current version of the ENDEC chip we propose either (I,T), (I,R) or (I,S). If future versions will allow a "L" to pass the pre-filtering function of course the (I,L) sequence defined in the standard can be programmed.

The hybrid mode does not need the PS-part all the time and it therefore must be possible to totally stop all PS-activities without loosing or changing the actual internal state. AMD has fortunately foreseen such a request and added two external accessible pins with nearly the necessary funtionality.

The whole design has been done with an integrated CAD/CAE-design system. It is based on a strictly hierarchical top-down concept with six layers of hierarchy in our case. The advantages of an integrated system and a consequent use of hierarchical design strategies are described in /11/.

Due to an incomplete CMOS-library of layouts and - more important - according models for the different layers of simulation a prototype printed circuit board (PCB) version based on discrete TTL-ICs and electrical programmable logic devices (EPLDs) was chosen first. To give an impression about the complexity : the number of ICs only for the slave H-MUX lies over 170 and needs at least 6 double Europe cards.

The partitioning and the necessary inter-board connections introduced a lot of problems. However, the placement and routing of 6 times about 30 circuits each was of course done much better and faster as one big design with 170 ICs.

Every component was chosen carefully. The timing was verified with worst case specifications of clock tolerances, component values and voltage level. Special care has been taken of the real-time critical parts of the design, as the evaluation and reprogramming of the control

bits, the electrical by-pass installation/removement and the internal synchronization of data with the according control signals.

Using a symmetrical two clock scheme data latching in succeeding stages becomes uncritical, but this also reduces the available time in a stage from 80 ns (byte oriented 100 Mbps equals 12.5 MBps and therefore 80 ns per byte) to ideally 40 ns and due to clock jitter only to practicable 30 ns. An 8bit comparator in fast AS-TTL-technology has a worst case propagation time of 25 ns, which shows how tough the constraints were.

Outlook

Two major topics are of further interest: The integration of this design into VLSI-ICs and the extension of the concept to provide access to future public networks.

The evaluation of the functionality and verification of the timing for a design with such a complexity is not possible purely by hand and on paper. So although we have a camera ready PCB-layout we are working on an expansion of the simulation module library of a VLSI development tool into two directions. First we want to simulate the functionality of the PCB-design, therefore a special TTL oriented library must be built up. At the same time we start to include CMOS-based modules with exactly the same functionality into a second library. When both libraries will be complete, a switch from the existing PCB-design towards a standard cell and macro based CMOS-layout should be possible with reasonable manpower.

First investigations have prooved the feasibility of current 2 micron CMOS technology for applications with these high speed requirements. There is even enough spare capacity to allow cell based design concepts instead of the optimized but time-consuming and error susceptible full custom design.

Intensive research activity will be focused on the concept and possible implementation of the station management and the general node processor architecture.

Future public networks must be able to deal with all existing and new services, with any of a range of characteristics. A network based on the Asynchronous Transfer Mode (ATM) is proposed by CCITT /12/ as the target network for Broadband-ISDN (B-ISDN), which will cope with these requirements. Since there is a need for the interconnection of HSLANs with public networks, a specific ATM-gateway is required.

The node concept presented in this paper can easily be extended to provide the functions necessary for integrating ATM traffic. Mainly, there are two possibilities /7, 10/: The ATM cells

can be transmitted in a dedicated ATM channel, which is composed of several WBCs. In this case the gateway comprises only the functionality of a repeater. The other possibility is to pack the ATM cells into synchronous packets. Here the gateway has to convert the ATM Virtual Channel Identifiers (VCIs) into FDDI MAC addresses.

For both possibilities an ATM interface is necessary which supports the following functions: Multiplexing and demultiplexing of ATM connections, transmission speed adaptation, synchronization and FDDI-I-MAC or P-MAC functions, respectively. In order to minimize the delay, the ATM interface should have direct access to the H-MUX.

Acknowledgement

We would like to thank H. Riethmüller for his engaged work concerning especially the H-MUX design.

References

/1/ FDDI Physical Layer, Draft International Standard 9314-1, 1987

/2/ FDDI Physical Layer Medium Dependent, Draft Proposed ANSI Standard X3T9.5, Rev. 7.3, 1988

/3/ FDDI Token Ring Media Access Control, ANSI Standard X3T9.5, 1988

/4/ FDDI Station Management, Draft Proposed ANSI Standard X3T9.5, Rev. 5, 1989

/5/ FDDI Hybrid Ring Control, Draft Proposed ANSI Standard X3T9.5, Rev. 5, 1989

/6/ IEEE 802.6/D7, Draft Proposed Standard Distributed Queue Dual Bus (DQDB) Metropolitan Area Network (MAN), May 1989.

/7/ Pohlit, R.; *ATM Transmission in FDDI-II*, Proc. EFOC/LAN 89, Amsterdam, The Netherlands, June 14-16, 1989, pp. 156-161.

/8/ Ross, F.E.; *FDDI - A Tutorial*, IEEE Comm. Mag., Vol. 24, No. 5, 1986, pp. 10-17.

/9/ *The SUPERNET Family for FDDI*; AMD databook, March 1989

/10/ Schödl, W.; Tangemann, M.; *Strategies for Interconnecting HSLANs to B-ISDN and their Performance*, to be published

/11/ Siegel, M.; Lemppenau, W.; *Integrierte VLSI-Entwicklungsumgebung für kommunikationsorientierte Schaltkreise* (in german), Mikroelektronik für die Informationstechnik, Stuttgart, 1989.

/12/ CCITT, Draft Recommendation I.121; *On the Broadband Aspects of ISDN*, COM XVIII-R55(C), Seoul, February 1988

/13/ Falconer, R.M.; *Carrying Integrated Services on the Orwell Slotted Ring*, ISonCNPE 85, Tokyo, 1985, pp. 4.11-4.11

/14/ Greaves, D. et. al.; *The Cambridge Backbone Network*, EFOC/LAN 88, Amsterdam, 1988